Automated Conformance Testing of C2IS

Michael Gerz / Nico Bau
Fraunhofer FKIE
Fraunhofer Institute for Communication, Information Processing, and Ergonomics
Neuenahrer Straße 20
53343 Wachtberg
Germany

Email: michael.gerz@fkie.fraunhofer.de / nico.bau@fkie.fraunhofer.de

Achim Vogt / Robert Vogt
ESG
Elektroniksystem- und Logistik-GmbH
Livry-Gargan-Straße 6
82256 Fürstenfeldbruck
Germany

Email: achim.vogt@esg.de / robert.vogt@esg.de

ABSTRACT

Command, Control, and Information Systems (C2ISs) are complex software products that must conform to agreed standards in order to be interoperable in joint and combined operations. Due to the complexity of information exchange between heterogeneous C2ISs, thorough testing is indispensable to gain confidence that these standards are implemented correctly and semantic interoperability is indeed achieved.

Testing can be performed in many different ways and by various means. For the Multilateral Interoperability Programme (MIP), a vendor-independent reference facility has been developed that checks the conformance of national C2ISs to the MIP specifications. MIP defines an interoperability standard for information exchange, which is of particular relevance for land forces and for information exchange between the services. The MIP Test Reference System (MTRS) is integral part of the MIP testing strategy. So far, the MTRS has been used for testing 27 C2ISs over the Internet.

In this paper, we introduce the concepts and features of the MTRS and describe an industrial perspective of using a conformance test system such as the MTRS. Moreover, we highlight our approach to testing mappings of C2IS information onto/from a common data model in the context of replication-based information exchange.

1.0 INTRODUCTION

A Command, Control, and Information System (C2IS) is a complex software product that must conform to agreed standards in order to be interoperable in joint and combined operations. An interoperability standard, which is of particular relevance for military forces, is the solution developed by the *Multilateral Interoperability Programme* (*MIP, [4]*). Meanwhile, 27 nations and NATO have joined this community of interest and even more nations are going to become a member of MIP.

While MIP develops a set of specifications that cover both operational and system aspects, implementing the MIP solution is a national issue. I. e., MIP defines a common interface for information exchange that

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
OCT 2009	**N/A**	**-**

4. TITLE AND SUBTITLE
Automated Conformance Testing of C2IS

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Fraunhofer FKIE Fraunhofer Institute for Communication, Information Processing, and Ergonomics Neuenahrer Straße 20 53343 Wachtberg Germany

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADB381582. RTO-MP-IST-087 Information Management - Exploitation (Gestion et exploitation des informations). Proceedings of RTO Information Systems Technology Panel (IST) Symposium held in Stockholm, Sweden on 19-20 October 2009., The original document contains color images.

14. ABSTRACT
Command, Control, and Information Systems (C2ISs) are complex software products that must conform to agreed standards in order to be interoperable in joint and combined operations. Due to the complexity of information exchange between heterogeneous C2ISs, thorough testing is indispensable to gain confidence that these standards are implemented correctly and semantic interoperability is indeed achieved. Testing can be performed in many different ways and by various means. For the Multilateral Interoperability Programme (MIP), a vendor-independent reference facility has been developed that checks the conformance of national C2ISs to the MIP specifications. MIP defines an interoperability standard for information exchange, which is of particular relevance for land forces and for information exchange between the services. The MIP Test Reference System (MTRS) is integral part of the MIP testing strategy. So far, the MTRS has been used for testing 27 C2ISs over the Internet. In this paper, we introduce the concepts and features of the MTRS and describe an industrial perspective of using a conformance test system such as the MTRS. Moreover, we highlight our approach to testing mappings of C2IS information onto/from a common data model in the context of replication-based information exchange.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**14**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

each individual C2IS needs to provide. The interface specification includes exchange mechanisms, a common data model, and operating procedures.

Some functionality of the MIP solution can be developed independently from a specific C2IS. In fact, some companies provide their MIP gateway implementation for several C2ISs. However, when it comes to the mapping of C2 information from the internal data model of a C2IS onto the common information exchange data model and vice versa, the tricky task of defining and implementing transformation rules has to be done for each C2IS individually.

Due to the complexity of C2 information exchange, thorough testing is indispensable. In order to identify potential problems with the specifications and/or the systems at an early stage, it is essential to start testing right from the start of the implementation. However, at this point in time, one's own C2IS may not be robust enough to perform interoperability tests efficiently with other systems.

For MIP Baseline 3, the Fraunhofer Institute for Communication, Information Processing, and Ergonomics (Fraunhofer FKIE) has developed a vendor-independent test facility that allows to test the conformance of C2ISs to the MIP specifications. Conformance testing is defined as functional black box testing. Unlike interoperability testing, which checks whether two or more systems are able to exchange information with each other, conformance testing aims at checking whether a single system meets a specific standard – such as the MIP specifications.

The test tool known as *MIP Test Reference System* (*MTRS*) has become integral part of the MIP testing strategy. So far, 27 C2ISs have been tested with the MTRS, including C2ISs that did not participate in the collocated interoperability tests conducted under the auspices of MIP. Using the MTRS, C2IS manufacturers and customers are able to execute tests over the Internet at any time (24x7) without the need to coordinate with other parties. Based on formalized test scripts, it is possible to gain reproducible and unbiased test results.

The MTRS has been described in several publications (e.g., see [2, 3]). In this paper, we focus on the industrial perspective of a conformance test system. On the technical level, we highlight our approach to test mappings of C2IS information onto/from a common data model in the context of replication-based information exchange.

This paper is structured as follows: In section 2, we provide a brief overview of the features of MIP baseline 3. Section 3 covers the MIP test process. In particular, we explain what it means to be semantically interoperable based on a common data model and show how semantic interoperability can be checked in the context of both interoperability and conformance testing. Section 4 sketches the most important features of the MIP Test Reference System. Based on a sample test case, we present our approach to test the mapping of C2IS user input onto MIP data packets and vice versa. In addition, we describe different ways to fully automate the testing process. In section 5, we look beyond the Multilateral Interoperability Programme and discuss to what degree the approach taken by the MTRS can be adapted to other military standards. Section 6 describes the benefits of a reference test system from an industrial perspective. Finally, section 7 provides a short summary.

2.0 THE MIP SOLUTION

The MIP solution enables information exchange between C2ISs and allows users to decide what information is exchanged, to whom it is distributed, and when. A high-level, conceptual view of the MIP solution is given in Figure 1.

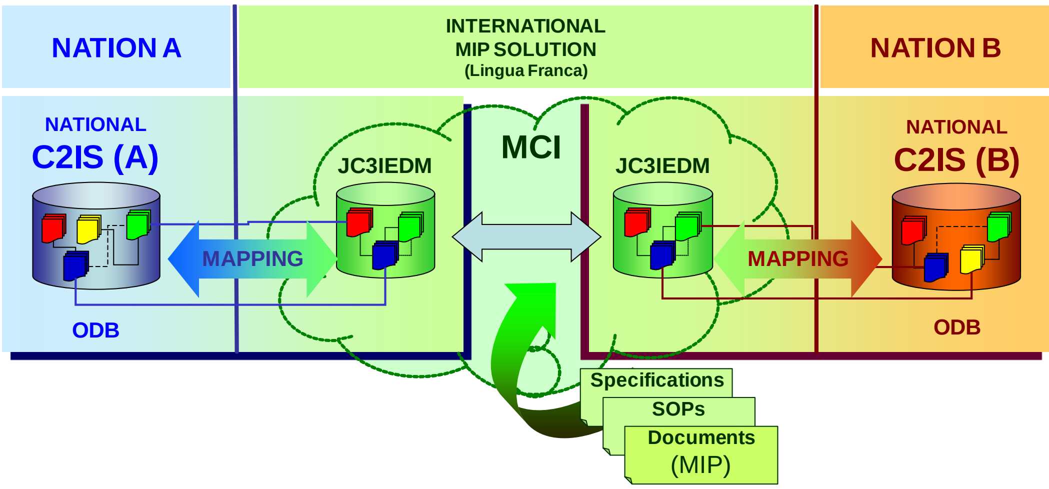


Figure 1: The MIP Solution (updated version of [5])

A core feature of the MIP solution is the *Joint Consultation, Command, and Control Information Exchange Data Model* (*JC3IEDM*), which is standardised as STANAG 5525. The JC3IEDM is an entity-relationship model in IDEF1X notation. It provides the basis for information exchange and specifies the semantics of militarily relevant objects, actions, etc., as well as the semantics of their relationships in an unambiguous way.

The logical view of the JC3IEDM consists of more than 270 entities, over 1400 attributes, and more than 230 relationships. Additionally, a huge number of business rules specify the correct usage of the data model. Most rules are stored along with the definition of all entities, attributes, relationships, and domain values in the *MIP Information Resource Dictionary* (*MIRD*). For a detailed description of the JC3IEDM and its extensions in Baseline 3, please see [1].

In addition to the data model, MIP defines information exchange protocols and procedures to achieve interoperability among heterogeneous C2ISs. The MIP *Data Exchange Mechanism* (*DEM*) supports the partial replication of operational data depending on their affiliation to a particular *Operational Information Group* (*OIG*). The DEM uses a publish-subscribe mechanism and includes an automated discovery service for local area networks (LAN). Thus, all systems in the network know about the presence of all other MIP-compliant C2ISs and are able to connect to them.

On the decision of the commander, his own OIGs can be shared with a partner on a case-by-case basis. Similarly, the commander may offer to forward operational information groups to which he is currently subscribed. Technically, if a potential data receiver opens a connection to a data provider, the latter replies with a list of all available OIGs. Subsequently, the receiver can subscribe to those OIGs he is interested in. Once he is subscribed to one or more OIGs, he will automatically receive all information updates of the respective OIGs. When the connection is closed, e.g., due to a network interruption, the receiver can resubscribe to the OIG. In that case, he can provide a synchronization point to indicate which information has been received successfully so far.

The DEM protocols support detailed error handling. In case of syntactic or semantic errors in the exchanged operational data, the data receiver may provide helpful information to the sender. This feature is particularly useful during testing. Since the DEM is based on database replication, exchanged data have to be referentially complete, i.e., they shall not violate foreign key constraints in the receiver's database.

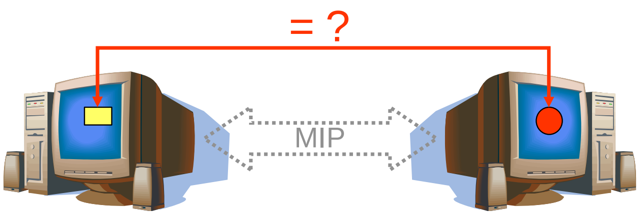


Figure 2: Interoperability Testing

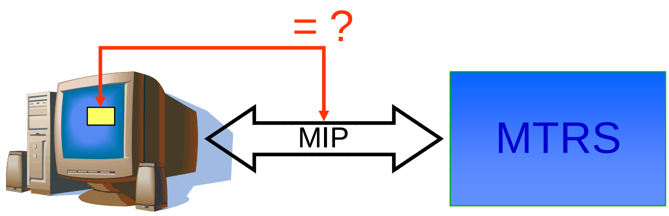


Figure 3: Conformance Testing

3.0 TESTING SEMANTIC INTEROPERABILITY

Testing the implementation of the MIP solution in national systems must cover all aspects of the MIP specifications. MIP distinguishes between system and operational tests. System-level tests check the functional properties of systems whereas operational-level tests validate the MIP solution (as implemented in the C2ISs) against its operational requirements.

There are three levels of system tests:

- Tests for the communication protocols (SLT1)
- Tests for database replication (SLT2)
- Tests for information exchange between C2ISs (SLT3)

SLT1 and SLT2 address aspects of the MIP solution that are typically implemented in a gateway component. The degree to which a MIP gateway is coupled with a C2IS varies among the different systems. As mentioned above, several vendors have developed MIP gateway products that are re-usable for multiple C2ISs.

Unlike SLT1 and SLT2, SLT3 covers end-to-end information exchange. In the context of interoperability testing, SLT3 checks that information created with C2IS *A* can be exchanged over the TCP/IP network and are displayed correctly by the receiving C2IS *B*.

Bilateral interoperability testing is sketched in Figure 3. To determine the test verdict, the test operators – typically one for each national system – have to compare the information presented at the user interfaces of their respective C2IS. Please note that the representations do not have to be identical – what matters is that the semantics of the information at C2IS *A* (as interpreted by test operator *TA*) is identical to the semantics of the information at C2IS *B* (as interpreted by test operator *TB*). For instance, two C2ISs may use different standards to represent tactical symbols. In order to assess SLT3 interoperability tests, the test operators do not need any knowledge about the technical details of the MIP solution. However, if a test fails, it is difficult to find out which of the two C2ISs is implemented erroneously.

When testing conformance with a test system, the mapping of user inputs onto MIP data packets and vice versa is checked. This approach is depicted in Figure 3. For conformance testing, there is no need for the test system to represent information on screen. Instead, the test system checks whether information created with the C2IS is mapped correctly onto the common data model of the interoperability standard and is exchanged according to the rules of the exchange mechanism.

When testing semantic interoperability, not all elements of the common data model need to be tested at the same level of detail. For instance, we can assume that free-text fields do not trigger any automated processes within a C2IS and that their representation in a C2IS is trivial. In contrast, some entities in the exchange data model may include attributes whose individual domain values are of utmost importance (such as the hostility status in the JC3IEDM). In the context of MIP, tactical symbols according to APP6a should be tested individually, because they are characterized by a combination of different attributes in the JC3IEDM. Moreover, MIP does not only specify the type of symbol but also its geometry. For instance, action *attack* is graphically represented by an arrow, which – according to another business rule – is modelled as a line with exactly two points in the JC3IEDM (where the first point denotes the basis and the second point represents the head of the arrow).

As said before, testing semantic interoperability means to test whether user input is processed and exchanged correctly as data packets (and vice versa), where the operational meaning of the data – as defined by the common data model – is identical to the semantics of the information provided by the C2IS user. I. e., a test case has to specify the required user action ('*Create XYZ…*') and the data that are expected to be exchanged. An important prerequisite is that there is a clear stimulus-response relationship between user input and exchanged data. A test case has to be specified and conducted in a way that its test verdict is not influenced by the former execution of another test case.

In the context of a replication-based information exchange, special technical aspects need to be considered for the test system:

- **References to formerly exchanged data –** Replication-based exchange aims at eliminating data exchange redundancy. That means that data may be transmitted only once even if they are referred to several times. For instance, in MIP, type information (on units, control features, etc.) is shared between objects of the same type. The type information itself is only transmitted for the first object. Therefore, it is not possible to simply analyse incoming protocol data units. Instead, the test system has to maintain a database that contains all (historic) data.
- **Unknown data elements –** Not all data received by the test system can actually be specified in advance. Some data is created dynamically at run-time by the C2IS itself and cannot be predicted. This holds, for instance, for database keys and timestamps. While the concrete value of a database key cannot be specified in a test case, it is important to specify the *relationships* between different database entities, i.e., the equality of keys in data records across multiple database tables.
- **Additional data elements –** There may also be operational data that are not relevant for the given test purpose but will be transmitted nevertheless. One reason is that data packets must be syntactically and semantically complete according to the rules of the exchange mechanism. Another reason is that a C2IS may cluster information internally (in terms of business objects) and replicate them as a collection. For instance, a C2IS may always exchange the location along with its unit even though the location was not requested. A test case must be specified and interpreted in a way that additional data does not automatically result in a failed test. A test case specification should list both required and forbidden data elements – all other data elements are considered as irrelevant.

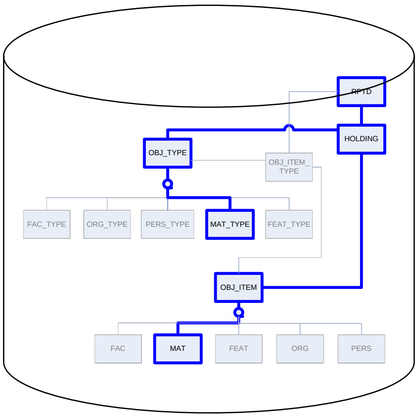


Figure 4: Identifying relevant entities in the database of the test system

- **Unknown stable testing state –** In the first step of a test case, the test system has to open a connection/subscription to the C2IS under test such that the C2IS is able to exchange its data with the test system. Once the connection/subscription is established, depending on the state of the C2IS and the history of information exchange between the C2IS and the test system, the C2IS may replicate some initial data. In case of the MIP solution, there is no way for the data provider to indicate that there are any pending data. Moreover, there is no assumption on the maximum amount of time it takes the data provider to collect, encode, and send data. In other words, it is not clear when a stable testing state is reached in which all the data have been exchanged that are not related to the test objective. For testing with the MTRS, we assume that data will be replicated within a timeframe of 20 seconds. If data have been received, the MTRS will wait for another 20 seconds, and so on. Unfortunately, this means that test execution is delayed for at least 20 seconds at the beginning of each test case.

In summary, a typical test case includes the following steps:

- The test system opens a connection/subscription to the C2IS under test.
- The C2IS may or may not send some initial data that are not directly related to the test purpose.
- The test system requests some user input that is performed by the test operator.
- The C2IS replicates some data using the standardized information exchange interface.
- The test system processes the received protocol data units and stores the operational data in its local database.
- The test system checks its database for the presence/absence of entities as specified in the test case (details are described in section 4.1), applying some pattern matching approach (see Figure 4).

4.0 CONFORMANCE TESTING WITH THE MIP TEST REFERENCE SYSTEM

The MIP Test Reference System (MTRS) has been developed to support the correct implementation of the MIP specifications in national C2ISs. It aims at decreasing the resources needed for testing MIP implementations while increasing their quality at the same time. The MTRS is complemented by a formal test suite for MIP SLT1, SLT2, and SLT3 that enables automated execution and evaluation of test cases on all system levels.

```
prompt 'Please create an organization named "Org-0630" with a materiel holding
                  (having an operational count greater than 0) in OIG FRDNEU.';
oig FRDNEU {
       new unique ORG org with name_txt = 'Org-0630';
       new unique HOLDING oldHolding;
       fixed MAT_TYPE matType;
       new fixed RPTD oldRptd;

       oldHolding -> org;
       oldHolding -> matType;
       oldHolding -> oldRptd;

       assert:
                 oldHolding.operat_cnt > 0;
}
prompt 'Please decrease operational count of the materiel holding.';
oig FRDNEU {
       new unique HOLDING newHolding;
       new RPTD newRptd;

       newHolding -> org;
       newHolding -> matType;
       newHolding -> newRptd;

       assert:
                 newRptd.rep_dttm > oldRptd.rep_dttm;
                 newHolding.operat_cnt < oldHolding.operat_cnt;
}
```

Figure 5: Sample test case for JC3IEDM mapping

The MTRS has been designed to meet the specific needs of a multinational interoperability programme. For instance, national test operators are informed on any relevant change to the test specifications that requires re-testing. Reporting capabilities have been integrated to support the MIP test controllers. The MTRS is offered free of charge to all nations participating in MIP.

The MTRS is a client-server application that allows to execute test cases over the Internet. For that purpose, the test operator uses a client application that allows him to view a comprehensive test suite and execute individual test cases. When running a test case, the MTRS server sets up one or two MIP gateways and connects to the C2IS under test.

The MTRS provides detailed logs and error reports in order to support the tester in error analysis. For instance, all interaction between the C2IS under test and the MIP gateway(s) on the MTRS server is logged and can be analyzed in a structured and human-readable representation. Automatically generated test reports make it possible for the MIP test organization and national controllers to track the progress of the systems.

4.1 JC3IEDM Mapping Tests

In the following, we describe the MTRS approach to test the mapping of C2 user input onto the JC3IEDM and vice versa.

4.1.1 C2IS as Data Provider

As described above, the test scripts for mapping tests should describe the expected state of the JC3IEDM database after a test (step) has been executed. For this, the MTRS uses a formal, declarative test language.

A sample test case is shown in Figure 5. It tests the ability of a C2IS to report on an organisation that holds materiels of some arbitrary kind and, in a second step, to report that the amount of assigned equipment has been reduced. The test checks that all required entities are received and linked correctly and the updated information has actually been reported later.

The SLT 3 test scripts used by the MTRS do not specify the detailed steps to establish a connection or to subscribe to an OIG, since these technical aspects of the exchange mechanism have been tested thoroughly in SLT 1 and SLT 2. In addition, it is not needed to specify checks for general replication rules. At run-time, the test system will check implicitly whether the C2IS conforms to the syntactic and semantic rules of the MIP solution. Therefore, the test specifier can focus on those aspects that are related to the test purpose.

Each JC3IEDM mapping test is specified as a sequence of test steps, each consisting of

- a user request,
- an OIG subscription, and
- a database check.

The test language uses *prompt* statements to describe the actions that the test operator has to perform. The MTRS client displays the corresponding text messages in a dialog box during test execution.

The subscription to an OIG is expressed by a single statement (Figure 5, lines 02 and 16). The test framework makes sure that all lower level communication between the test system and the C2IS takes place. In case of multiple test steps with different OIGs, the MTRS will also unsubscribe from a former OIG, where needed.

Lines 03 to 06 in Figure 5 describe the relevant JC3IEDM entities. The keyword *new* indicates that these entities have to be received *after* the *prompt* command has been executed, the *with* clause adds an additional check on the entity's attribute. *new* and *with* work as local constraints, i.e., they restrict the search pattern on a single entity.

In a first validation step, the MTRS checks its database for matches of each required entity, including the provided local constraints. If such a check is successful, and the database contains one or more records that match the search pattern, the uniqueness of entities is checked. All entities declared as *unique* must have only one match in the database. If more matching records are found, the test fails. If no matching entity exists, the test system waits for more data.

When one or more matching records are found for all required entities, the MTRS tries to identify the whole structure. Therefore, it interprets the lines 08 to 10 in Figure 5 as associations between the records. Internally, the MTRS maps these associations to a single SQL statement that joins all required entities and includes all local constraints. If this query does not produce any result, the test system waits for more data again. Otherwise, the result is separated into the different entities, i.e., the result set is split in a way that each entity described in the test step is mapped to one or more data records of the result set.

Next, the MTRS checks that only one record has been found for all entities that are marked as *fixed*. In contrast to the *unique* keyword, the *fixed* keyword is called a global constraint, as it takes into account all other constraints that are described in the test step. In our example, this means that the JC3IEDM database

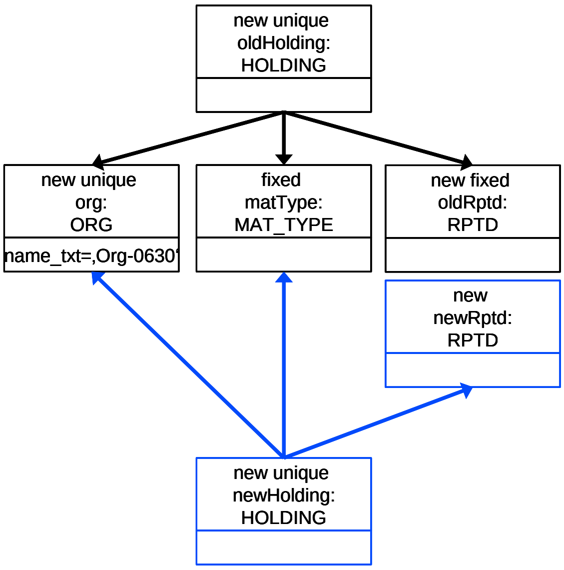


Figure 6: Graphical representation of expected data structure

may contain many MAT_TYPEs, but only one MAT_TYPE that is linked to a HOLDING, which again is linked to an ORG with name *Org-0630* that was received during test execution (see Figure 6). Although test scripts may look very simple, the resulting SQL queries can get very complex. If a result has been found and all *unique* or *fixed* entities were found only once, then the MTRS checks all assertions on the retrieved data (line 13).

After a test step has been executed successfully, the MTRS keeps the data records of all required entities and continues with the next test step if there is any. Again, the MTRS can automatically determine the required actions it needs to perform to subscribe to the correct OIG (line 16 in Figure 5).

Because of the formal description of the test cases, it is possible to generate coverage information and to generate graphical representations like the one shown in Figure 6.

4.1.2 C2IS as Data Receiver

Each JC3IEDM mapping test offered by the MTRS can also be run in the opposite direction, i.e., with the C2IS receiving data. Producing test data (in terms of data base records) for 350+ test cases is an error-prone and arduous task. Rather than specifying test data by hand, we implemented a tool that collects data provided by C2ISs using the MTRS.

Whenever a C2IS passes an SLT 3 test (as data provider), the MTRS asks the test operator whether he is willing to share the data with other nations. The test operator has the choice to share the data with the C2IS name attached to it, to share that data anonymously or not to share the data at all.

If the operator decides to share the data, the MTRS queries its database and stores all data related to the test case in a format that can be used to send them as DEM-conformant PDUs. Thus, the test operator is now able to receive data that was produced by his system or any other system that shared its data for the specific test case. After the C2IS has successfully received the data, the MTRS gives the test operator a detailed description of what he should see on screen. The test operator has to confirm this.

4.2 Test Automation

To facilitate regression tests and tests at an early stage of development, it is beneficial to reduce the cost of testing to a minimum. This can be done by automating the test process. As we have already described, the MTRS server can run and evaluate a test case without the need of manual interaction on the server side. This is because the formal test scripts describe the test events unambiguously and therefore they can be executed automatically.

However, on the C2IS side, there is no standardized interface to perform tests. The only interface specified is the MIP DEM, and this protocol is not designed to facilitate testing. Therefore, the test operator has to use the MTRS Client to start, stop, and analyze tests. Since a test normally consists of different steps such as opening a connection, receiving or sending information, etc., the MTRS Client asks the test operator to perform different tasks with his C2IS to produce the desired behaviour.

One way to automate this process is to use a GUI automation tool, which can simulate user actions such as pressing a key or moving or clicking the mouse. For each test case, the required actions that the test operator has to perform can be recorded and replayed automatically during regression testing.

This approach can be used for both the C2IS under test as well as the MTRS Client. However, it has several limitations:

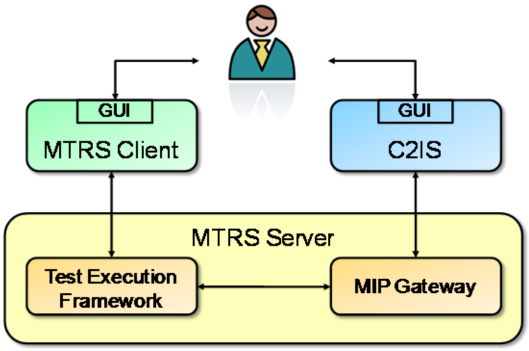
- First, it relies on the behaviour of the MTRS Client and the C2IS being deterministic. All events have to occur in exactly the same order and (depending on the capabilities of the automation tool) with roughly the same delay. While the order of events of a single test case is defined by the test script and therefore stable (unless the test script is changed), the C2IS may perform internal actions that might result in a different order and timing of events. This problem can be solved by adding an additional notification into the C2IS's control flow that indicates when an event has occurred.
- Second, for each test case there has to be a unique script that simulates the necessary user input to either the C2IS or the MTRS Client. This makes it very hard to include conditional inputs or to follow alternative control flows in case an error occurred. Furthermore, it does not allow to reuse parts of the script that occur in many test cases, such as opening a connection.

One advantage of using a GUI automation tool to stimulate the C2IS is that the input of the automation tool closely mimics the test operator's input, making the test more realistic by testing the whole C2IS from the GUI to the MIP interface.

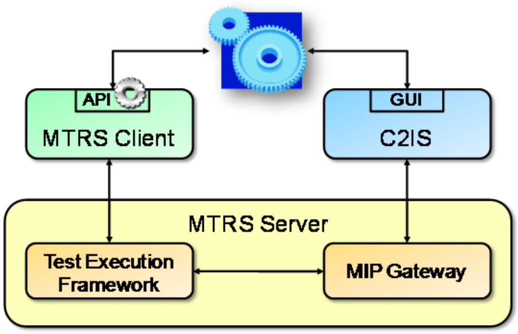
In order to help national implementers to further automate their testing process, an MTRS API can be used to control the MTRS Server. The MTRS API provides simple methods to log into the MTRS Server and to start and stop test cases. Furthermore, it supports callback methods that can be used to process events (required user interactions as well as detailed logging information). This enables both a simple automation of the test cases and detailed error analysis and error handling (e.g., a test can be aborted automatically if a timer expires).

If the C2IS also has the capability of being controlled by an external program (or using a GUI automation tool as discussed above), it is possible to completely automate the testing process, allowing the implementers to build an infrastructure that can be used for – even distributed – continuous testing during the whole development phase.

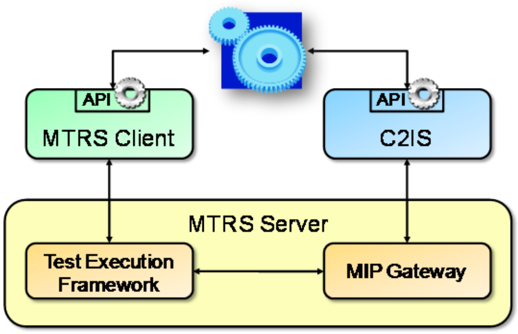
The API has been used in practice by one MIP nation to fully automate all SLT1 and SLT2 test cases, i.e., all test cases related to the DEM protocols and to database replication. For that purpose, they have set up a distributed test environment that runs up to four test cases in parallel with the same number of C2IS instances. This enables them to re-run all test cases whenever their system or the test specifications change. So far, they have run more than 10,000 tests!



(a) Semi-automatic test execution



(b) Fully automatic test execution with GUI tester



(c) Fully automatic test execution with C2IS-specific test interface

Figure 7: Test Execution

5.0 ADAPTATION TO OTHER MILITARY STANDARDS

The utility of the MIP Test Reference System has been appreciated by the MIP Programme Management Group. Due to the positive feedback received from the test operators, we are confident that other military standardization projects will benefit from a similar reference facility. Potential candidates include standards like APP-11/ADatP-3, NFFI, MAJIIC, Link, or OTH-T Gold.

In principle, the conformance testing approach can be applied to any standard for which the expected behaviour of the systems under test can be specified – at least partially. In case of the MIP solution, there are no assumptions on the concrete processing and presentation of information. However, as a common denominator, we can assume that the C2IS user is able to view and interpret information received from a remote C2IS (or the MTRS). If systems process and evaluate information in an automated manner – which is the case, e.g., for constructive simulation systems – it is harder or even impossible to specify the expected outcome independently from a specific system.

Technically, most of the core concepts, architecture, and software components of the MTRS are independent from a specific standard and a specific test suite. For instance, the MTRS supports a general-purpose test language that enhances Java with control structures for testing distributed systems. They allow to describe alternative behaviour, timers, and exceptions in a declarative manner. The SLT3 test notation shown in section 4.1 is mapped onto this core language. Thus, it can be considered as a MIP-specific extension of the test infrastructure. Likewise, the MTRS client has (almost) no knowledge about the MIP solution and provides key features for error-diagnosis, reporting, and test versioning independently from the MIP implementation details.

Existing military information exchange standards differ with regard to the coverage of their specifications. For instance, APP-11 provides a catalogue of message text formats, most of which are specified in accordance with the structure rules defined by ADatP-3. ADatP-3 also specifies formatting rules for the physical exchange of messages. Starting with ADatP-3, Baseline 13.1, message text formats can be expressed as XML schemas and exchanged as XML documents, too. Unlike MIP, APP-11 and ADatP-3 do not prescribe a specific exchange mechanism. In order to support automated conformance testing, one or more commonly agreed exchange mechanisms would have to be selected and (partly) implemented in the test system. However, testing APP-11 itself is restricted to testing the semantics of individual message text formats. (Please note that error handling, i.e., the reaction to invalid messages – violating syntax and static semantics – is a feature of the exchange mechanism!)

Test cases for APP-11 would be conceptually similar to the MIP SLT3 test cases outlined above, where the system under test is either data sender or data receiver:

In the first case, the test operator has to input some operational data that is encoded and sent as APP-11 message by the C2IS. The test system will check the well-formedness of the message received, i.e., its syntax and static semantics, and compare the content of all relevant fields with the expected values specified in the test case.

In the second case, the test system sends a predefined message to the C2IS under test. The C2IS meets the test objective if the test operator confirms that some specific information (encoded in the message) is presented correctly, i.e., the semantics is preserved. Test cases that are more complex check the processing and presentation of consecutive messages with, e.g., information updates and conflicting information. The aggregation of data from messages of different types should also be subject to testing (please note that APP-11 does not have an underlying common data model in general).

From a technical perspective, testing the mapping of APP-11 messages is simpler than testing the mapping of JC3IEDM data elements, because messages are typically self-contained. Database constraints, such as referential integrity, do not have to be taken into account.

Unlike APP-11, NFFI brings along three protocol stacks for reliable and unreliable communication. Information is encoded as XML messages. In order to specify test cases on a proper level of abstraction, parts of the protocol stacks would have to be implemented in the test system. Apart from that, standard approaches to protocol conformance testing can be applied.

6.0 INDUSTRIAL PERSPECTIVE

To illustrate the advantages of a test reference system, we will provide a brief overview of how testing was done without the MTRS. Testing should be started as soon as possible to ensure interoperability from early on. Unfortunately, multinational tests have several drawbacks, which make them not a good option for early tests:

- First, different nations have different timeframes and priorities for their implementations, making it nearly impossible to find a testing partner during the early development phase.
- Second, due to different time zones, tests might have to be conducted outside the regular working hours.
- Third, there is a big organisational overhead to schedule a testing session between multiple nations. To make matters worse, if an error is encountered during a testing phase, this might result in the need to abort the session because the error prevents further testing.
- Last, but not least, a multinational test involves human test operators on both sides. Since the test procedure and test documents as well as the evaluation of the test results are somewhat open to interpretation, this might lead to situations where both test operators wrongly assume that their implementation is correct.

Because of these difficulties, the only viable way to conduct tests at an early stage without a test reference system is testing locally. There are two possible approaches: One way of testing involves generating test data manually and sending it to the gateway. The reaction of the C2IS (e.g., display of a unit) has then to be verified manually against the expected result. Unfortunately, creating the test data is a rather difficult and time-consuming task.

The second approach consists of using two instances of the same implementation to test with each other. The two systems might be connected using a local area network. This enables the tester to create test data in one C2IS and to look at the output of the other. However, errors in the mapping of C2IS data onto the JC3IEDM might not be noticed by this approach. Furthermore, it is impossible to test the C2IS's reaction to invalid data since the C2IS is (or should be) unable to produce such data.

With a conformance test system, the overhead of organizing multinational test sessions can be delayed to a later point in development, when all systems have proven their conformance. Then, the testers can focus more on the operational side of the specifications, knowing that the technical side has been tested thoroughly.

The MTRS is available to all implementers all the time (even outside standard business hours). Since the server side is fully automated, there is no need for coordination; all tests can be conducted as soon as they are available. The availability of more than 1,400 test cases for SLT1, SLT2, SLT3, and Symbology for MIP Baseline 3 enhanced the software quality management. Additionally, there is no room for interpretation as all tests are fully formalized and reviewed/validated by all nations. This does not mean that all tests are correct by default; the writer of a formal test case still might have made an error but the interpretation and evaluation is fair and consistent for all systems, allowing all nations to report an error in a test script.

One disadvantage of the automated test execution is the lack of flexibility; all test steps have to be performed exactly as formalized. This might lead to rather tedious repetition of some basic steps such as connecting the C2IS to the MTRS server, exchanging basic information, and so on. In section 4.1, we have described two ways of automating these tasks on the client side.

Although the MTRS provides some “general purpose” test cases, where the test system behaves as a simple MIP gateway and checks all received information for conformance to the MIP specifications, it would also be desirable to be able to execute user-generated test cases.

The detailed logging capabilities of the MTRS allow the test operator to review the information flow within the MTRS, providing the capability to analyse errors. As each test result is stored and all test results are aggregated for each test group, a colour-coded icon next to each test group gives a convenient overview of the current test status. Additionally, the MTRS Client informs the test operator of changes to previously conducted test cases, allowing the test operator to re-run these tests. All test results, including information about the test duration, the test operator etc. can be printed as a test overview. This removes the burden of filling out a multitude of test reports – as it was necessary for MIP Baseline 2 – from the test personal.

From the perspective of the industry, the major advantage of the MTRS is the significant reduction of time and costs in testing the national implementation of the MIP Baseline 3 solution.

7.0 SUMMARY

In this paper, we have presented the core concepts of the MTRS with focus on the tests of operational data. Although the official MIP tests for MIP Baseline 3 have been finished, the MTRS is still used on a daily basis by several systems.

The MTRS provides more than 630 system-level tests for functional black box testing, including more than 350 mapping tests (in either direction!). In addition, there are hundreds of symbology tests, which can be used to test the mapping of APP6a symbols. Overall, more than 70,000 test runs have been executed by the MTRS so far.

One important factor that made the MTRS a success story is its independence from any specific vendor. Since the MTRS solely focuses on testing, it does not compete with any commercial C2IS. Moreover, all test cases have been developed in cooperation with the MIP community or are subject to public review.

8.0 REFERENCES

[1] Bau, N., Gerz, M., Glauer, M., & Schüller, H. (2007b). Multilateral Interoperability Programme – Advancements in MIP Baseline 3. In Twelfth International Command and Control Research and Technology Symposium (12th ICCRTS) Newport, RI, USA: Office of the Assistant Secretary of Defense, Command and Control Research Program (CCRP).

[2] Bau, N., Gerz, M., & Glauer, M. (2008a). Ensuring interoperability of command and control information systems – new ways to test conformance to the MIP solution. Journal of Telecommunications and Information Technology, (2), 5-13.

[3] Bau, N., Gerz, M., & Glauer, M. (2008b). Testing C2 Interoperability – Advancements in Testing of the MIP Baseline 3 Solution. In: Proceedings of the 13th International Command and Control Research and Technology Symposium (ICCRTS), Bellevue, WA, USA.

[4] Multilateral Interoperability Programme (2009). MIP website. www.mip-site.org.

[5] Multilateral Interoperability Programme (2006). MIP Standard Briefing. http://www.mip-site.org.